

REMARKS

The Office Action and the references cited therein have been carefully considered. Claims 1-35 are pending, remain pending in this application and are at issue herein. Claims 15, 16, and 35 have been amended to correct typographical errors. Specifically, the word "and" has been added to claim 15 before the last claim element and a semi-colon has been added to the end of the first element of claim 16. Claim 35 has been amended to add the word "to" between "function" and "disconnect" in line of claim 35. These amendments do not change the scope of the claim.

The Examiner has rejected Claims 1 - 35 under 35 U.S.C. 103(a) as being unpatentable over Man (U.S. Patent No. 5,710,908) in view of what the Examiner calls Applicants Admitted Prior Art. The Applicant respectfully traverses this ground of rejection. Reconsideration of these rejections in view of the following comments is respectfully solicited.

The Examiner states that Man teaches a computer readable medium having instructions for performing the steps for directing data transfer in a computer having a plurality of transport providers (network protocols) the steps comprising: receiving transport specific data (data packet/transmission packet) from an application (SNMP manager) (col. 11, lines 49-53; col. 14 lines 15-20); determining a transport (protocol; protocol stacks) with which the transport specific data is associated (via the protocol address mapping table) (col. 12, lines 6-39; col. 14, lines 15-35); and passing the transport specific data to the transport (sending the data packet using the selected transport) (col. 13, lines 42-50; col. 14, lines 15-35).

The Applicants respectfully disagree. The Examiner is directed to col. 10, lines 18-line 45, and col. 12 lines 6-39 combined with col. 13 lines 42-53. Man teaches a protocol independent interface (PII) that, when initialized, determines which transport protocols are available for use, assigns an access line to each available transport protocol and an access ID to each program such as the SNMP manager, and creates mapping information that indicates a one-to-one correspondence between the access ID/access line pair and a block of transport protocol specific information. When a data packet is sent to the PII with the access ID of the sending program (e.g., the SNMP manager), a transport protocol is selected to transmit the data packet. A default or preferred protocol is selected if available. For example, UDP is the preferred protocol for transmitting data between a SNMP manager and a SNMP agent. After selecting the

transport protocol to transmit the data packet, the PII retrieves a block of transport protocol specific information based on the transport protocol selected (corresponding to the access line) and the access ID of the transmitting program (e.g., the SNMP manager). The PII then forms a transmission packet that is specific to the selected protocol that contains the block of transport specific information and the data packet. The transmission packet contains transport specific data (e.g., IPX, UDP, or DDP headers). The transmission packet is then sent to the destination application program via the transport protocol selected. The multiplexer software module 220, to which the Examiner refers to, serves as a multiplexer which routes packets between network interface driver 210 and the protocol stacks (i.e., IPX, UDP, DDP).

The Applicants have thoroughly reviewed Man and can find no teaching, suggestion, or motivation to receive transport specific data from an application, determine at least one of the plurality of transport modules with which the transport specific data is associated, pass the transport specific data to the at least one of the plurality of transport modules, and send a transport independent interface to the application as required by claim 1. As previously indicated, Man receives protocol independent data, selects a transport protocol to use, and then forms a transport specific packet for sending the data via the transport protocol that was selected after having received the protocol independent data.

The Applicants respectfully submit that selecting a transport protocol and then forming a transport specific transmission packet that contains data from an application is not the same as receiving transport specific data from an application and then selecting a transport module with which the transport specific data is associated. Furthermore, if the data of Man were transport specific, then the principle of operation of Man would change as Man is directed to a protocol independent method of transmitting data. Such a modification to a reference is not allowed (See MPEP 2143.01).

Additionally, Man teaches away from the present invention. Specifically, Man teaches that if a preferred protocol is not available for data that is received, then the first available protocol is used. Alternatively, randomly selected protocols may be selected. Only transport modules associated with the transport specific data can be used in the instant invention. The instant invention returns an error message to the application if there are no transport modules installed that are associated with the transport specific data from an application. Therefore, not only does Man not teach or suggest receiving transport specific

data from an application and then selecting a transport module with which the transport specific data is associated, Man teaches away from receiving transport specific data from an application and then selecting a transport module with which the transport specific data is associated.

The Examiner also states that Man teaches that since no information indicating a transmission protocol is required for the packet, the packet is protocol independent and a single interface is provided between an application and all protocol stacks (col. 11, lines 53-57, col. 13 line 54 - col. 14, line 42) and that it would be obvious that a transport independent interface is given to the application in order to perform the other functions. The Examiner also states that Man teaches that a SNMP agent (application) is also returned a table of entry points that the SNMP agent uses to call other PII routines when the PII is initialized and that it would therefore be obvious that the application has a transport independent interface.

It is improper, in determining whether a person of ordinary skill would have been led to a combination of references, simply to use that which the inventor taught against its teacher. The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 16 USPQ2d 1430 (Fed. Cir. 1990). Further, conclusory statements cannot be relied upon when dealing with particular combinations of prior art and specific claims. The rationale on which the Examiner is relying upon must be set forth. *In re Lee* 61 USPQ2d (Fed. Cir. 2002). The Applicants respectfully submit that the Examiners statements are conclusory statements because the Examiner has not put forth sufficient reasons for modifying the Man reference. For example, the statement of "it would have been obvious to one of ordinary skilled in the art at the time the invention was made to recognize that such transport independent interface was given to the application in order to perform the other functions" is a conclusory statement. Man has been reviewed and it does not teach or suggest providing a transport independent interface to an application. Additionally, the Examiner has merely stated that it is obvious that the application has a transport independent interface based on a table of entry points that a SNMP agent uses to call other PII routines when the PII is initialized. It is respectfully submitted that a transport independent interface is not needed when a table of entry points is provided. Therefore, it would not be obvious to

provide a transport independent interface to perform the other functions, which the Examiner has not defined, based on the Man reference.

Therefore, for the reasons set forth above, it is respectfully requested that the Examiner remove the 35 U.S.C. 103(a) rejection of claim 1.

Claims 2-11 depend from claim 1 and are believed to be patentable for the same reasons put forth above for claim 1. Furthermore, in the Examiner's rejection, Examiner states that what the Examiner calls Applicants Admitted Prior Art teaches: the OBEX application, that each of the plurality of transport modules has a transport protocol, and that at least one protocol is one of an IrDA protocol, an IP protocol, and Bluetooth protocol. The Examiner then states that it would be obvious to incorporate the teachings of what the Examiner calls Applicants Admitted Prior Art into the Man's teaching in order to modify the SNMP manager or SNMP agent to perform the OBEX application functions.

SNMP is an abbreviation that stands for Simple Network Management Protocol, which is a set of protocols for managing complex networks. In simple terms, it is a popular network management protocol commonly used with TCP/IP: SNMP lets TCP/IP-based network management clients exchange detailed information about their configuration and status. In SNMP, agents, which can be hardware as well as software, monitor the activity in the various devices on the network and report to the network console workstation. Object Exchange (OBEX), on the other hand, is a compact binary protocol that enables a wide range of devices to wirelessly exchange data spontaneously in a simple, efficient manner. OBEX is primarily used as a push or pull application and is similar to HTTP (Hypertext Transfer Protocol) but does not require the resources that an HTTP server requires, making OBEX ideal for devices with limited resources. OBEX consists of an application framework and a transport protocol, which consists of a format for communication between devices and applications and a set of functions.

A transport protocol does not function as a network management protocol. Modifying a management protocol to perform a transport protocol would render the SNMP protocol unsatisfactory for its intended use and change the principle of operation of the Man reference, which is not allowed per MPEP 2143.01. Specifically, Man teaches that "if an application program must support multiple network protocols, duplicated effort is required for the application software to handle the different APIs. For example, an SNMP program must

include software code to communicate with an IPX protocol stack and a DDP protocol stack, in addition to code to communicate with a UDP protocol stack." Therefore, the Applicants respectfully submit that one would not combine the teachings of Man with the teachings of what the Examiner calls Applicants Admitted Prior Art in order to modify the SNMP manager or SNMP agent to perform the OBEX application functions.

With respect to claims 7-11, the Examiner states that neither Man nor what the Examiner calls Applicants Admitted Prior Art detail how the transport interface comprise detail commands such as: initializing a transport, creating a connection, enumerating devices, enumerating properties, closing the transport, closing the connection, listening for incoming connection, etc. and that it would have been obvious to one of ordinary skill in the art at the time the invention was made to recognize that such commands or operations would have been a matter of programming choices for interface between two devices or systems. The Applicants respectfully disagree.

It is well known that to establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. In the present application, the Examiner states that neither Man nor what the Examiner calls Applicants Admitted Prior Art details what the transport interface comprises but that it would have been a design choice to have the commands or operations as claimed in claims 7-11. Man (and what the Examiner calls Applicants Admitted Prior Art) has been thoroughly reviewed and no teaching or suggestion could be found to have the commands/operations of claims 7-11 nor is there any teaching or suggestion of a transport independent interface. Furthermore, the Examiner has not put forth any evidence to show that the commands or operations of claims 7-11 are design choices.

Therefore, for the reasons set forth above, it is respectfully requested that the Examiner remove the 35 U.S.C. 103(a) rejection of claims 2 - 11.

With respect to claims 12-35, the Examiner states that the claims are similar in scope as claims 1-11 and therefore are rejected for the same reasons set forth above for claims 1-11. The Applicants respectfully submit that the Examiner must show that the Man reference and what the Examiner calls Applicants Admitted Prior Art teach or suggest all of the elements of claims 12-35 and that there is some suggestion or motivation to combine the references. The teaching or suggestion must be found in the prior art and not based on the Applicant's disclosure. The Examiner has not put forth any reasons to support the statement that the claims are similar in scope. Furthermore, MPEP 706.02(j) requires that the Examiner should set forth the relevant teachings of the prior art relied upon, preferably with reference to the relevant column or page number(s) and line numbers(s) where appropriate, the difference or

differences in the claim over the applied reference(s), the proposed modification of the applied reference(s) to arrive at the claimed subject matter, and an explanation of why one of ordinary skill in the art at the time the invention was made would have been motivated to make the proposed modification. It is respectfully submitted that the Examiner has not taken any of these steps in the rejections of claims 12-35.

Furthermore, independent claim 12, for example requires the steps of creating a primary interface, finding at least one of a second device, connecting to the at least one of a second device through a device interface, and commanding one of a put command and a get command to transfer at least one object between a first device and the at least one of a second device. The primary interface includes a command to enumerate transports, a command to enumerate devices, and a command to register a service. The device interface includes a connect command to connect to a device, a put command to put an object on a device, a get command to get an object from a device, a command to disconnect a connection, a command to abort a request, and a command to set a path. Man and what the Examiner calls Applicants Admitted Prior Art have been thoroughly reviewed and neither Man nor what the Examiner calls Applicants Admitted Prior Art teach or suggest all of the elements of claims 12 to 16. For example, no teaching or suggestion of creating a primary interface or commanding one of a put command and a get command to transfer at least one object between a first device and at least one of a second device could be found. Additionally, Man teaches that the PII creates a mapping table and uses the mapping table to connect to devices. Man does not provide a device interface to transfer objects to a second device as required by claim 12.

Independent claim 17 requires providing a service to at least one device comprising the steps of listening for an incoming connection, receiving a service connection interface when an incoming connection is received, the service connection interface for listening for incoming command requests, listening for incoming command requests from the at least one device, receiving a command structure when an incoming command request is received that describes the incoming command request, and performing one of a read and a write operation in response to the incoming command request. Dependent claim 18 requires creating a primary interface having a register command to register a service, reading a transport data blob from a registry, passing the transport data blob to the register command, and receiving a service interface from the primary interface to listen for an incoming connection. The service connection interface includes a command to accept an incoming connection, a command to close a connection, a command to listen for incoming connections, and a command to get the properties of a connection. The

command structure includes a pointer to an interface to enumerate headers that were received with a connect request, a command to generate a response code, and a stream interface to use to interface with a data stream. The stream interface includes a command to read data from a stream, a command to write data to the stream, a command to read data from a specified file, and a command to write data to the specified file. Man and what the Examiner calls Applicants Admitted Prior Art have been thoroughly reviewed and neither Man nor what the Examiner calls Applicants Admitted Prior Art teach or suggest all of the elements of claims 17 to 21. For example, no teaching or suggestion of receiving a service connection interface or a command structure could be found. The PII of Man listens for packets addressed to specific sockets, e.g., sockets defined for use by management programs, and ignores all others. Listening for a packet addressed to a socket is not the same as listening for an incoming connection as required by claim 17.

Independent claim 22 requires a computer readable medium having computer-executable instructions for performing steps to provide at least one service to at least one device through at least one transport, the steps comprising providing a primary interface, the primary interface having a command to enumerate transports and to enumerate devices, providing a transport interface for communicating with the at least one transport, providing a service interface for determining when an incoming connection arrives, and providing a device interface for communicating with the at least one device.

Dependent claim 23 requires that the primary interface comprises a function to enumerate transports, a function to enumerate devices, and a function to register a service. Dependent claim 24 requires that the transport interface comprises a function to initialize a transport, a function to create a socket, a function to enumerate a list of devices of a specified type, a function to enumerate properties required to create a listening socket, and a function to close a transport. Dependent claim 25 requires providing a transport socket interface if a socket is created. Dependent claim 26 requires that the transport socket interface comprise a function to close a socket, a function to listen for incoming connections, a function to enumerate properties about a socket, and a function to connect to at least one of the at least one device. Dependent claim 27 requires providing a transport connection interface if at least one of the at least one device is connected. Dependent claim 28 requires that the transport connection interface comprise a function to close a connection, a function to send data on the connection, a function to receive data on the connection, and a function to enumerate properties about the connection.

Dependent claim 29 requires that the service interface comprises a function to listen for an incoming connection for the at least one service, a function to shut down an

instance of the at least one service, and a function to set a password required to access the at least one service. Dependent claim 30 requires providing a service connection interface if the incoming connection comes in. Dependent claim 31 requires that the service connection interface comprises a function to accept a connection, a function to close the connection, a function to listen for at least one incoming command request from the at least one of the at least one device, and a function to enumerate properties of the connection. Dependent claim 32 requires providing a command structure if the at least one incoming command request is received. Dependent claim 33 requires that the command structure comprises a pointer to an interface to enumerate at least one header that came in with the incoming connection, a function to generate a response code, and a stream interface. Dependent claim 34 requires that the stream interface comprises a function to read data from a stream, a function to write data to the stream, a function to instruct the stream to use data from a specified file, and a function to instruct the stream to write data to the specified file. Dependent claim 35 requires that the device interface comprises a function to connect to a device, a function disconnect the device, a function to send data to the device, and a function to get data from the at least one service.

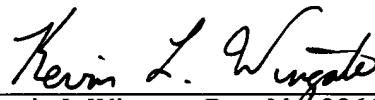
Man and what the Examiner calls Applicants Admitted Prior Art have been thoroughly reviewed and neither Man nor what the Examiner calls Applicants Admitted Prior Art teach or suggest all of the elements of claims 22 to 35. For example, no teaching or suggestion of providing a primary interface, the primary interface having a command to enumerate transports and to enumerate devices, providing a transport interface for communicating with the at least one transport, providing a service interface for determining when an incoming connection arrives, and providing a device interface for communicating with the at least one device could be found. As previously indicated, the PII of Man listens for packets addressed to specific sockets, e.g., sockets defined for use by management programs, and ignores all others. It does not listen for an incoming connection. Man provides no interface for listening for a packet addressed to the socket. Furthermore, listening for a packet addressed to a socket is not the same as determining when an incoming connection arrives as required by claim 22 because the connection has already been established.

Therefore, since neither Man nor what the Examiner calls Applicants Admitted Prior Art teach or suggest all of the elements of claims 12-35, it is respectfully submitted that a *prima facie* case of obviousness has not been made. Therefore, for the reasons set forth above, it is respectfully requested that the Examiner remove the 35 U.S.C. 103(a) rejection of claims 12 - 35.

In re Appln. of Eric Davison et al.
Application No. 09/587,753

The application is considered in good and proper form for allowance, and the Examiner is respectfully requested to pass this application to issue. If, in the opinion of the Examiner, a telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney.

Respectfully submitted,



Kevin L. Wingate, Reg. No. 38662
LEYDIG, VOIT & MAYER, LTD.
6815 Weaver Road, Suite 300
Rockford, Illinois 61114-8018
(815) 963-7661 (telephone)
(815) 963-7664 (facsimile)

Date: September 9, 2003